VERACODE

VOLUME 12

The State of Software Security Industry Snapshot: Financial Services

Veracode's State of Software Security (SOSS) Volume 12 examines historical trends shaping the software landscape and how security practices are evolving along with those trends. The data collected from 20 million scans across half a million applications suggests that we're making good progress toward the goal of producing more secure software.

This SOSS snapshot provides a view of software security in the financial services sector. We hope it brings the findings a little closer to home so you can better refine your application security (AppSec) program based on the most relevant data. Let's start things off with Figure 1, which provides some core comparative metrics for the state of software security in the financial industry.

Starting on the left, the financial sector's overall flaw percentage ranks among the "best" (though most apps still have flaws) but falls to the middle of the pack for high-severity flaws. The industry is tied for the lowest proportion of those flaws that are fixed, though the percentages show little variation among industries. Overall, the first three columns suggest that many financial firms would benefit from better finding and fixing the flaws that matter most.

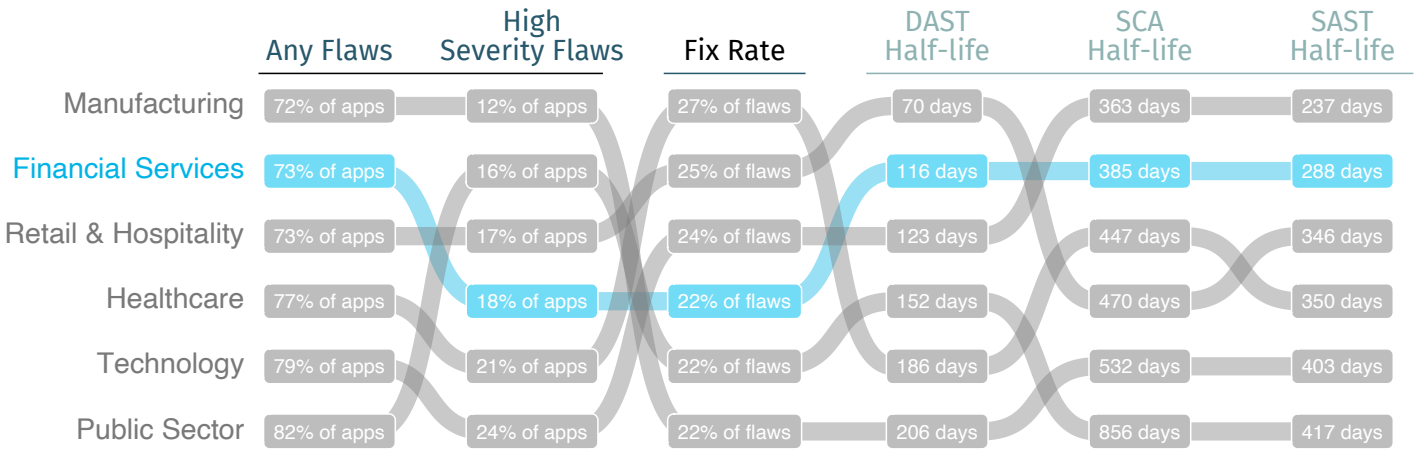

Figure 1: Values and rankings for key software security metrics by industry.

The rightmost columns rank industries according to how quickly they fix flaws once they're detected by three different types of scans. The financial sector boasts quick fix times for flaws discovered by static (SAST), dynamic (DAST), and software composition analysis (SCA) scans. That's a laudable accomplishment, but the number of days required to get to the halfway point shows there's still ample room for continued improvement.

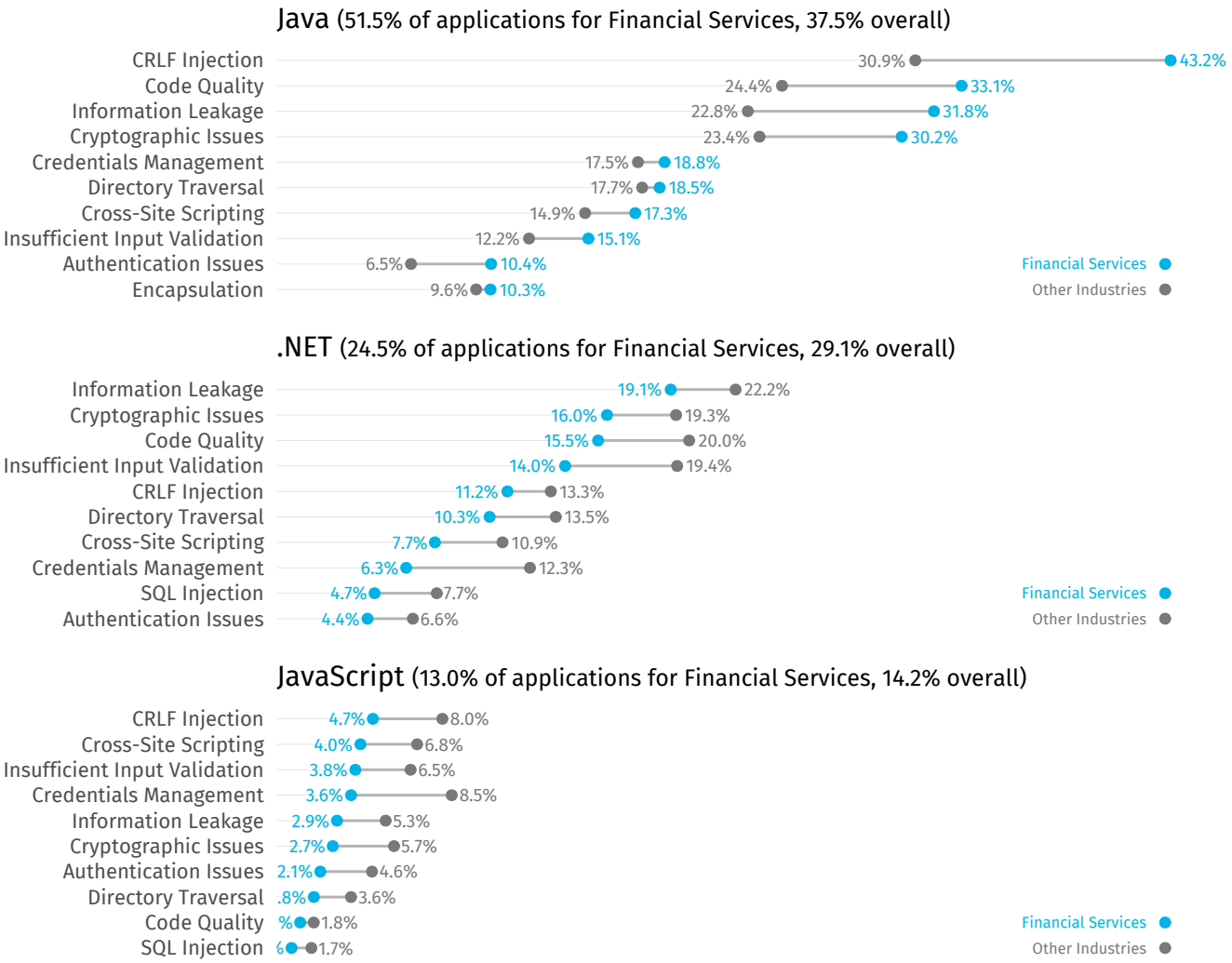

Figure 2: Most common flaws from static analysis in the finance sector.

Having compared overall flaw and fix rates, let's take a look at the most common types of flaws affecting applications. Because flaws found by SAST are very language-dependent, Figure 2 separates results by the top three programming languages used among applications in the finance sector. The chart makes it easy to determine whether financial firms (in blue) have higher or lower rates than the overall average (in gray) for each type of flaw. Results are mixed here, with finance scoring better than par for .NET and JavaScript apps and subpar for Java. There's a lot of information to digest, so we'll leave you to develop your own takeaways.

Unlike SAST, DAST findings are largely consistent across languages, leading us to combine the findings into one chart. Finance follows a similar pattern to that of other industries in terms of which flaws are commonly vs. rarely identified by dynamic analysis. The percentages for the financial sector are higher across the board, however, perhaps due to greater functional complexity inherent to financial applications.

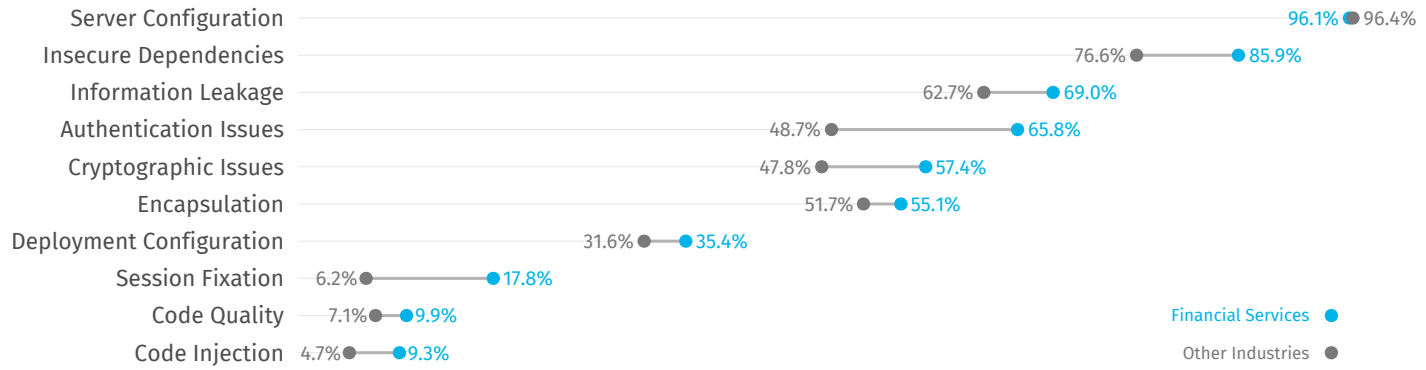

Figure 3: Most common flaws from dynamic analysis in the finance sector.

Next, we'll offer a few charts that expand on the half-life stats presented back in Figure 1. The number of days required to fix half the flaws in an application is a simple, benchmark-worthy stat, but what if you're curious about the comprehensive lifecycle of software security issues? Good news – Figure 4 enables exactly that using a method known as survival analysis!

Triangulating any point along the survival curve gives the percentage of flaws still "alive" after a period of time following discovery (e.g., over 40 percent still unresolved after one year). For the most part, the lifecycle of application flaws within financial firms tracks closely with other sectors. It's hard to distinguish the two survival curves for SAST. Finance is quick out of the gate addressing flaws found via DAST, but loses that lead around the one-year mark.

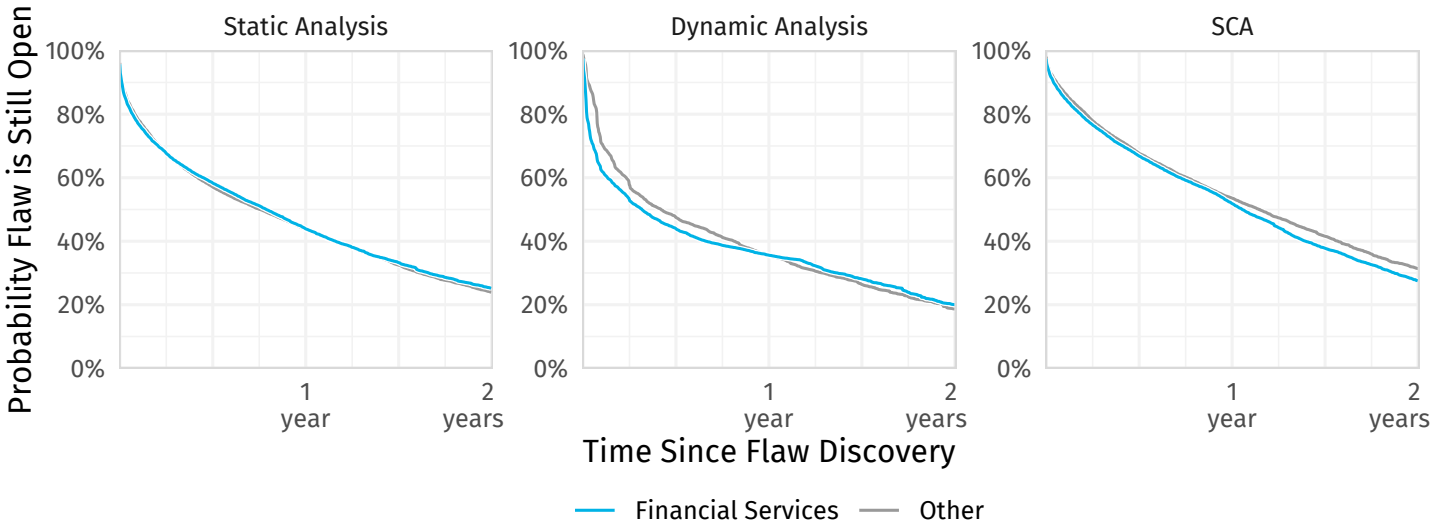

Figure 4: Two-year flaw survival rates for applications in the finance sector.

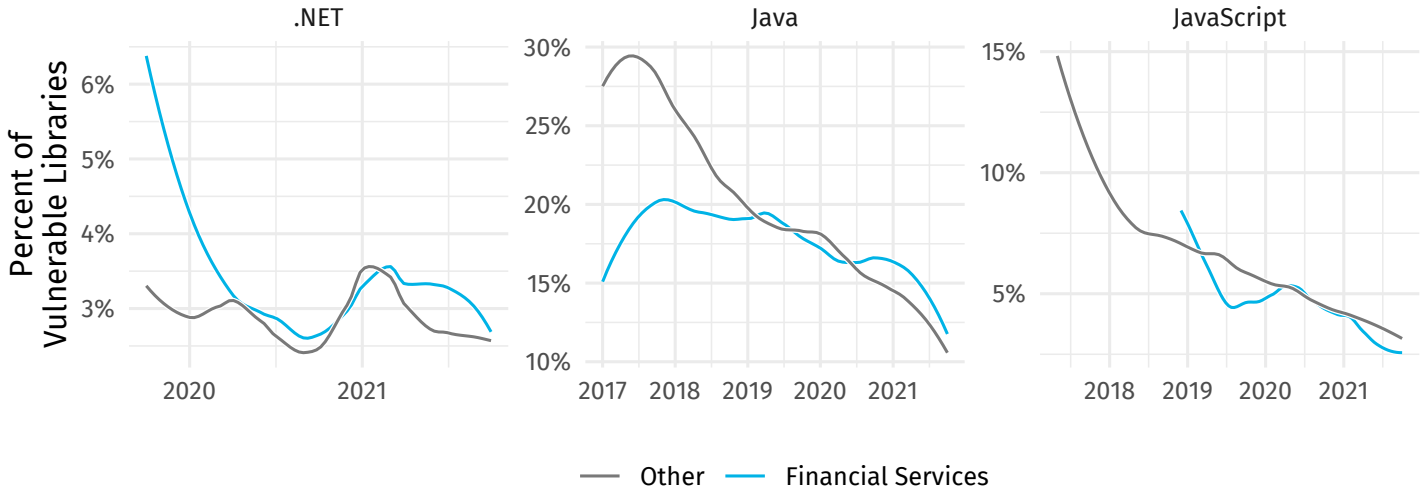

Figure 5: Proportion of vulnerable libraries used by applications in the finance sector.

Flaws in third-party libraries found through SCA stick around longer for all industries, with about 30 percent still unresolved after two years. The finance sector tows the line for the first year, but quickens the pace for addressing vulnerable libraries after that to gain about a month on the cross-industry average.

Speaking of vulnerable libraries, you're probably aware that the software supply chain is kind of a big deal these days among software security professionals. The last set of charts in this snapshot shows the extent of flaws in third-party code discovered via SCA. Java applications exhibit the highest ratio of vulnerable libraries, but that's trending down over time. The same can be said for the other languages, which is a welcome ray of sunshine in an otherwise gloomy realm of software security. Here's to increasingly clear skies in the years to come!

VERACODE

Veracode is a leading AppSec partner for creating secure software, reducing the risk of security breach, and increasing security and development teams' productivity. As a result, companies using Veracode can move their business, and the world, forward. With its combination of process automation, integrations, speed, and responsiveness, Veracode helps companies get accurate and reliable results to focus their efforts on fixing, not just finding, potential vulnerabilities.

Learn more at **www.veracode.com**, on the **Veracode blog**, on **LinkedIn**, and on **Twitter**.

Read the Full Report